Amendments to the Claims

Please amend Claims 1, 14, 39, 51, 63, and 75. The Claim Listing below will replace all prior versions of the claims in the application:

Claim Listing

1.   (Currently Amended) A method of managing memory for a plurality of data accessors in a computer having a plurality of memory blocks in a common system memory, comprising:

storing data in a first memory block in the common system memory of the computer, the plurality of memory blocks in the common system memory shared by the plurality of data accessors;

assigning a non-modifiable global view of the first memory block data to a first data accessor and a second data accessor in the computer;

in response to a request by the second data accessor to modify the first memory block, copying the data from the first memory block to a second memory block in the common system memory without regard to the existence of the data in any other memory block; and

assigning a modifiable private view of the second memory block data to the second data accessor in the computer so that a modification to the second memory block data by the second data accessor is not communicated to the first data accessor.

2.   (Original) The method of Claim 1 wherein the first and second data accessors are user process threads in a multi-threaded computer system.

3.   (Original) The method of Claim 1 wherein the memory blocks are a cache memory area.

4.   (Original) The method of Claim 1 further comprising associating a pointer accessible by the first and second data accessors to the first memory block.

5.	(Original) The method of Claim 1 wherein the second data accessor includes exclusive write access to the second memory block.

6.	(Original) The method of Claim 5 further comprising associating a private pointer accessible only by the second data accessor to the second memory block

7.	(Original) The method of Claim 1 wherein the first data accessor and the second data accessor read the first memory block without having exclusive access to the first memory block.

8.	(Original) The method of Claim 1 wherein storing comprises retrieving a specific generation of the data from a database.

9.	(Original) The method of Claim 1 further comprising selecting each memory block from a list of free memory blocks.

10.	(Original) The method of Claim 9 further comprising maintaining the list of free memory blocks by triggering a collector to free allocated memory blocks in response to a triggering event.

11.	(Original) The method of Claim 10 wherein the triggering event occurs when the number of free memory blocks falls below a threshold.

12.	(Original) The method of Claim 1 wherein the modifiable private view is a copy of the global view of data modifiable by the second data accessor.

13.	(Original) The method of Claim 1 wherein at least two data accessors independently access the non-modifiable view of the first memory block.

14.	(Currently Amended) A method of simultaneously analyzing data stored in a common system memory by multiple data accessors in a computer, comprising:

retrieving data from storage into public blocks of a shared memory space in the common system memory of the computer, the shared memory space in the common system memory shared by the data accessors, the public blocks storing data therein for global read-only access by a plurality of user accessors;

in response to receiving requests from user accessors to modify public blocks, copying public blocks of data into private blocks of the shared memory space, each private block storing data therein for private read and write access by a single user accessor without regard to the existence of the data by any other memory block;

upon read access to a data item by a user accsessor, reading the data item if present from a private block accessible by the user accessor, and if the data item is not present on a private block accessible by the user accessor, reading the data item from a public block; and

upon write access to the data item by the user accessor, writing the data item to a private block without communicating any modification to the private block to another user accessor.

15. (Original) The method of Claim 14 wherein the user accessors are computing threads in a multi-threaded computer system.

16. (Original) The method of Claim 14 wherein the shared memory space is a page cache.

17. (Original) The method of Claim 14 wherein writing the data item to a private block comprises copying data from a public block to the private block.

18. (Original) The method of Claim 14 further comprising associating each block of data with a generation of data from a database.

19. (Original) The method of Claim 14 further comprising:
    storing a set of public pointers accessible to the user accessor; and
    assigning each public pointer to a respective public block.

20. (Original) The method of Claim 14 further comprising:

        in a user accessor, storing a set of private pointers only accessible to the user accessor; and

        assigning each private pointer to a respective private block of the user accessor.

21. (Original) The method of Claim 14 wherein at least two data accessors independently access the non-modifiable view of the first memory block.

22. (Original) The method of Claim 14 wherein the modifiable private view is a copy of the global view of data modifiable by the second data accessor.

23. (Original) The method of Claim 14 wherein the user accessors read the read-only public blocks without having exclusive access to the public blocks.

24. (Original) The method of Claim 14 wherein storing comprises retrieving a specific generation of the data from a database.

25. (Original) The method of Claim 14 further comprising selecting each memory block from a list of free memory blocks.

26. (Original) The method of Claim 25 further comprising maintaining the list of free memory blocks by triggering a collector to free allocated memory blocks in response to a triggering event.

27. (Original) The method of Claim 26 wherein the triggering event occurs when the number of free memory blocks falls below a threshold.

28. (Previously Presented) An apparatus for managing a common system memory for a plurality of data accessors in a computer comprising:

a plurality of memory blocks in the common system memory shared by the plurality of data accessors, a first memory block and a second memory block having data stored therein;

a first data accessor executing in the memory, the first data accessor having a non-modifiable global view of the first memory block data; and

a second data accessor executing in the memory, the second data accessor having a non-modifiable global view of the first memory block data and a modifiable private view of the second memory block data, in response to a request by the second data accessor to modify first memory block data, so that a modification to the second memory block data by the second data accessor is not communicated to the first data accessor.

29. (Original) The apparatus of Claim 28 wherein the first and second accessors are user processing threads in a multi-threaded computer system.

30. (Original) The apparatus of Claim 28 wherein the plurality of memory blocks comprises a cache memory area.

31. (Original) The apparatus of Claim 28 wherein the second memory block has data copied from the first memory block.

32. (Original) The apparatus of Claim 28 further comprising a pointer accessible by the first and second accessors which references the first memory block.

33. (Original) The apparatus of Claim 28 wherein the second accessor has exclusive write access to the second memory block.

34. (Original) The apparatus of Claim 33 further comprising a private pointer accessible only by the second process which references the second memory block.

35. (Original) The apparatus Claim 28 wherein the data stored in the first memory block is a specific generation of the data retrieved from a database.

36.   (Original) The apparatus of Claim 28 wherein the first and second accessors select each memory block from a list of free memory blocks.

37.   (Previously Presented) The apparatus of Claim 36 further comprising a collector executing in the computer, the collector maintaining the list of free memory blocks by freeing allocated memory blocks in response to a triggering event.

38.   (Previously Presented) The apparatus of Claim 37 wherein the triggering event occurs when the number of free memory blocks falls below a threshold.

39.   (Currently Amended) An apparatus for simultaneously analyzing data stored in a common system memory by multiple data accessors in a computer comprising:

        public blocks of a shared memory space in the common system memory of the computer, the memory shared by the plurality of data accessors, the public blocks storing data retrieved from storage therein for global read-only access by a plurality of user accessors;

        private blocks of the shared memory space, each private block storing data therein for private read and write access by a single user accessor, the stored data including data selectively copied from public blocks of data in response to request by the single accessor to modify a public block;

        a read operation to provide read access to a data item by a user accessor, the read operation reading the data item if present from a private block accessible by the user accessor, and if the data item is not present on a private block accessible from the user accessor, reading the data item from a public block; and

        a write operation to provide write access to the data item by the user accessor, the write operation writing the data item to a private block so that a modification to the data item is not communicated to the other user accessors.

40.   (Original) The apparatus of Claim 39 wherein the user accessors are computing threads in a multi-threaded computer system.

41.   (Original) The apparatus of Claim 39 wherein the shared memory space is a page cache.

42.   (Original) The apparatus of Claim 39 wherein the private block comprises data copied from a public block to the private block.

43.   (Original) The apparatus of Claim 39 wherein each block of data is associated with a generation of data from a database.

44.   (Original) The apparatus of Claim 39 further comprising a set of public pointers accessible to the user accessor, wherein each public pointer is assigned to a respective public block

45.   (Original) The apparatus of Claim 39 further comprising a user process with a set of private pointers only accessible to the user accessor, wherein each private pointer is assigned to a respective private block of the user accessor.

46.   (Original) The apparatus as claimed in Claim 39 wherein the user accessors read the read-only public blocks without having exclusive access to the read-only public block.

47.   (Original) The apparatus of Claim 39 wherein the data stored in the first memory block is a specific generation of the data from a database.

48.   (Original) The apparatus of Claim 39 wherein the first and second accessors select each memory block from a list of free memory blocks.

49.   (Original) The apparatus of Claim 48 further comprising a collector executing in the computer, the collector maintaining the list of free memory blocks freeing allocated memory blocks in response to a triggering event.

50.   (Original) The apparatus of Claim 49 wherein the triggering event occurs when the number of free memory blocks falls below a threshold.

51.   (Currently Amended) An article of manufacture comprising:

a computer-readable medium;

a computer-implementable program stored in the computer-readable medium for execution by a computer, the program including:

storing data in a first memory block in a common system memory, memory blocks in the common system memory of the computer shared by a plurality of data accessors;

assigning a non-modifiable global view of the first memory block data to a first data accessor and a second data accessor in the computer;

in response to a request by the second data accessor to modify the first memory block, copying the data from the first memory block to a second memory block in the common system memory if the data is to be modified by the second data accessor, without regard to the existence of the data in any other memory block; and

assigning a modifiable private view of the second memory block data to the second data accessor in the computer so that a modification to the second memory block data by the second data accessor is not communicated to the first data accessor.

52.   (Original) The article of Claim 51 wherein the first and second data accessor are programmed as user threads in a multi-threaded computer system.

53.   (Original) The article of Claim 51 wherein the memory blocks are programmed as a cache memory area.

54.   (Original) The article of Claim 51 wherein the program further comprises reading the data stored in the first memory block from a multi-dimensional database.

55.   (Original) The article of Claim 51 wherein the program further comprises associating a pointer accessible by the first and second data accessors to the first memory block.

56. (Original) The article of Claim 51 wherein the second data accessor is programmed to include exclusive write access to the second memory block.

57. (Original) The article of Claim 56 wherein the program further comprises associating a private pointer accessible only by the second data accessor to the second memory block.

58. (Original) The article of Claim 51 wherein the first data accessor and the second data accessor read the first memory block without having exclusive access to the first memory block.

59. (Original) The article of manufacture of Claim 51 wherein storing comprises retrieving a specific generation of the data from a database.

60. (Original) The article of manufacture of Claim 51 wherein the program further comprises selecting each memory block from a list of free memory blocks.

61. (Original) The article of manufacture of Claim 60 wherein the program further comprises maintaining the list of free memory blocks by triggering a collector to free allocated memory blocks in response to a triggering event.

62. (Original) The article of manufacture of Claim 61 wherein the triggering event occurs when the number of free memory blocks falls below a threshold.

63. (Currently Amended) An article of manufacture comprising:
     a computer-readable medium;
     a computer-implemented program in the computer-readable medium for execution by a computer for simultaneously analyzing data stored in a common system memory by multiple data accessors in a computer, the program comprising:
          retrieving data from storage into public blocks of a shared memory space in the common system memory, the <u>shared memory space in the</u> common system

memory shared by the data accessors, the public blocks storing data therein for global read-only access by a plurality of user accessors;

selectively copying public blocks of data into private blocks of the shared memory space, each private block storing data therein for private read and write access by a single user accessor in response to a request by the single user accessor to modify the public blocks without regard to the existence of the data in any other memory block;

upon read access to a data item by a user accessor, reading the data item if present from a private block accessible by the user accessor, and if the data item is not present to a private block accessible by the user accessor, reading the data item from a public block; and

upon write access to a data item by the user accessor, writing the data item to a private block without communicating any modification to the data to another user accessor.

64. (Original) The article of Claim 63 wherein the user accessors are programmed as computing threads in a multi-threaded computer system.

65. (Original) The article of Claim 63 wherein the shared memory space is programmed as a page cache.

66. (Original) The article of Claim 63 wherein writing the data item to a private block comprises copying data from a public block to the private block.

67. (Original) The article of Claim 63 wherein the program further comprises associating each block of data with a generation of data from a database.

68. (Original) The article of Claim 63 wherein the program further comprises:
storing a set of public pointers accessible to the user accessors; and
assigning each public pointer to a respective public block.

69.   (Original) The article of Claim 63 wherein the program further comprises:

in a user accessor, storing a set of private pointers only accessible to the user accessor; and

assigning each private pointer to a respective private block of the user accessor.

70.   (Original) The article of Claim 63 wherein the user accessors read the read-only public blocks without having exclusive access to the public blocks.

71.   (Original) The article of manufacture of Claim 63 wherein retrieving comprises retrieving a specific generation of the data from a database.

72.   (Original) The article of manufacture of Claim 63 wherein the program further comprises selecting each memory block from a list of free memory blocks.

73.   (Original) The article of manufacture of Claim 72 wherein the program further comprises maintaining the list of free memory blocks by triggering a collector to free allocated memory blocks in response to a triggering event.

74.   (Original) The article of manufacture of Claim 73 wherein the triggering event occurs when the number of free memory blocks falls below a threshold.

75.   (Currently Amended) A method of managing memory for a plurality of data accessors in a computer having a plurality of memory blocks in a common system memory, comprising:

storing data in a first memory block in the common system memory, the memory blocks in the common system memory shared by the plurality of data accessors;

assigning a non-modifiable global view of the first memory block data to a first data accessor and a second data accessor in the computer;

associating a pointer accessible by the first and second data accessors to the first memory block;

in response to a request by the second data accessor to modify the first memory block, copying the data from the first memory block to a second memory block in the common system memory without regard to the existence of the data in any other memory block, the second data accessor including exclusive write access to the second memory block;

associating a private pointer accessible only by the second data accessor to the second memory block;

selecting each memory block from a list of free memory blocks;

maintaining the list of free memory blocks by triggering a collector to free allocated memory blocks in response to a triggering event; and

assigning a modifiable private view of the second memory block data to the second data accessor in the computer so that a modification to the second memory block data by the second data accessor is not communicated to the first data accessor.

76. (Original) The method of Claim 75 wherein the first and second data accessors are user process threads in a multi-threaded computer system.

77. (Original) The method of Claim 75 wherein the memory blocks are a cache memory area.

78. (Original) The method of Claim 75 wherein the first data accessor and the second data accessor read the first memory block without having exclusive access to the first memory block.

79. (Original) The method of Claim 75 wherein storing comprises retrieving a specific generation of the data from a database.

80. (Original) The method of Claim 75 wherein the triggering event occurs when the number of free memory blocks falls below a threshold.

81. (Original) The method of Claim 75 wherein the modifiable private view is a copy of the global view of data modifiable by the second data accessor.

82.   (Original) The method of Claim 75 wherein at least two data accessors independently access the non-modifiable view of the first memory block.